

**AMENDMENTS TO THE CLAIMS**

1. (Currently Amended) A ~~computer~~ system, comprising:  
a deterministic network;  
a computer executing a hard real-time operating system, said computer being  
connected to the deterministic network;  
an application running under the hard real-time operating system; and  
a security process running under the hard real-time operating system; and  
an external monitor connected to the deterministic network, wherein  
the security process is configured to periodically, in hard real-time, check the  
integrity of the application and/or a data element used by the application and, if the  
integrity check of the application or the data element indicates that the application or data  
element has been tampered with, notify a user of the computer system and/or shut down  
at least part of the computer system or application, and  
the security process includes a challenge handler that is configured to (i) receive a  
challenge transmitted from an-the external monitor to the challenge handler via the  
deterministic network and (ii) transmit to the external monitor via the deterministic  
network a response to the challenge in less than about five millisecond from the challenge  
handler receiving the challenge~~provide a response thereto within a predetermined amount~~  
of time, wherein the external monitor is configured so that if the external monitor does  
not receive the response within ~~a predetermined amount of time~~ five milliseconds or less  
from sending the challenge, the external monitor issues a notification ~~notifies an~~  
~~administrator and/or shuts down at least part of the computer system or application.~~

2. (Currently Amended) In a computer system running a real-time operating system, a computer security method, comprising:

executing a security process under the real-time operating system, wherein the security process is configured to periodically, in hard real-time, check the integrity of an application and/or a data element used by the application and ~~notify a system administrator~~ issue a notification and/or shut down the application if the integrity check of the application or the data element indicates that the application or data element has been tampered with;

sending, from an external monitor, a challenge to the security process or to a challenge handler that monitors the integrity of the security process via a deterministic network; and

sending to the external monitor via the deterministic network a response to the challenge, wherein the response is sent in less than about five milliseconds from when the challenge was received; and

issuing a notification and/or shutting down at least part of the computer system or the application notifying an administrator if a response to the challenge is not received within about five milliseconds or less from when the challenge was sent ~~a predetermined amount of time.~~

3. (Original) A computer system, comprising:  
a dual-kernel operating system comprising a real-time kernel and a non-real-time kernel;

a first real-time thread running under the real-time kernel, the first real-time thread being configured to monitor the integrity of an application running under the non-real-time kernel;

a second real-time thread running under the real-time kernel, the second real-time thread being configured to monitor integrity of the first real-time thread; and

a security process running under the non-real-time kernel, the security process being configured to check the integrity of the first real-time thread and/or the second real-time thread.

4. (Currently Amended) The ~~computer~~-system of claim 1, wherein the integrity check performed by the security process includes checking ~~the~~ an execution ~~scheduling schedule~~ of the application.

5. (Currently Amended) The ~~computer~~-system of claim 4, wherein the security process is configured to raise an alarm if, after checking the execution ~~scheduling schedule~~ of the application, the security process determines that the application is not being scheduled at a required minimum frequency.

6. (Currently Amended) The ~~computer~~-system of claim 1, wherein the integrity check performed by the security process includes checking the integrity of the application's code.

7. (Currently Amended) The ~~computer~~ system of claim 6, wherein the security process is configured to raise an alarm if, after checking the integrity of the application's code, the security process determines that the application code has been tampered with.

8-13. (Canceled)

14. (Currently Amended) The ~~computer~~ system of claim 1, wherein the security process is further configured to update a data item with a sequence number indicating a number of cycles that have passed without detection of an intruder.

15. (Currently Amended) The ~~computer~~ system of claim 14, wherein the security process is further configured to transmit the data item to the external monitor using an encryption key included in a challenge sent to the challenge handler.

16. (Currently Amended) The ~~computer~~ system of claim 15, wherein the security process is further configured to transmit the data item to the external monitor within a predetermined amount of time from when the external monitor sent a challenge to the challenge handler.

17. (Currently Amended) The method of claim 2, wherein the integrity check performed by the security process includes checking the an execution scheduling schedule of the application.

18. (Previously Presented) The method of claim 17, further comprising the step of raising an alarm in response to the security process determining that the application is not being scheduled at a required minimum frequency.

19. (Previously Presented) The method of claim 2, wherein the integrity check performed by the security process includes checking the integrity of the application's code.

20. (Previously Presented) The method of claim 19, further comprising the step of raising an alarm in response to the security process determining that the application's code has been tampered with.

21-24. Cancelled.

25. (Currently Amended) The method of claim [[24]] 2, further comprising the step of sending an encryption key to the security process at or about the same time as sending the challenge to the security process.

26. (Currently Amended) The method of claim 25, further comprising the steps of receiving the encryption key and encrypting the response using the encryption key prior to transmitting the response.

27. (Previously Presented) The computer system of claim 3, wherein the first real-time thread is further configured to check a set of integrity markers of the non-real-time kernel.

28. (Previously Presented) The computer system of claim 27, wherein the integrity markers include a checksum and/or digital signature of a data element that maintains information about a password file used by the non-real-time kernel.

29. (Previously Presented) The computer system of claim 28, wherein the data element is an inode.

30. (Previously Presented) The computer system of claim 28, wherein the application is programmed to encrypt and decrypt passwords stored in the password file.

31. (Previously Presented) The computer system of claim 3, wherein the second real-time thread is further configured to check a set of integrity markers of the real-time kernel.

32. (Previously Presented) The computer system of claim 3, further comprising a challenge handler executing under the real-time kernel.

33. (Previously Presented) The computer system of claim 32, further comprising an external monitor.

34. (Previously Presented) The computer system of claim 33, wherein the challenge handler is responsive to challenges sent from the external monitor to the challenge handler.

35. (Previously Presented) The computer system of claim 34, wherein the challenge handler is configured to send a response to the external monitor in response to receiving from the external monitor a challenge.

36. (Previously Presented) The computer system of claim 35, wherein the response includes an encrypted data item.

37. (Previously Presented) The computer system of claim 35, wherein the external monitor is programmed to determine whether the response from the challenge handler was received by the external monitor within a predetermined amount of time.

38. (Previously Presented) The computer system of claim 37, wherein the external monitor is further programmed to raise an alarm if it determines that the response from the challenge handler was not received by the external monitor within the predetermined amount of time.

39. (New) The system of claim 1, wherein the challenge handler is configured to provide a response within about one millisecond.

40. (New) The system of claim 1, wherein the security process is configured at system boot with a periodicity to check the integrity of the application.

41. (New) The system of claim 1, wherein the response is encrypted.

42. (New) The method of claim 2, wherein the challenge handler is configured to provide a response within about one millisecond.

43. (New) The method of claim 2, wherein the security process is configured at system boot with a periodicity to check the integrity of the application.

44. (New) The method of claim 2, wherein the response is encrypted.